AMENDMENTS TO THE SPECIFICATION

Please replace paragraph [0042] with the following amended paragraph:

[0042] If the master component 110 is not asserting anything on the data wire 132 during the preamble phase, then the data wire 132 should carry a logical one if none of the slave components is transmitting the remainder of a prior frame on the data wire 132. Alternatively, even if the master component 110 may be asserting a logical one on the data wire during at least some of the preamble, then the data wire 132 should still be carrying the logical one during the preamble phase assuming that none of the slave components is transmitting on the data wire 132 at that time. On the other hand, the frame is designed such that neither a master nor a slave transmits more than fifteen consecutive logical ones in a row when transmitting non-preamble ~~none-preamble~~ portions of the frame.

Please replace paragraph [0045] with the following amended paragraph:

[0045] Therefore, the preamble is significantly shortened while further retaining error recovery from loss of synchronization. Furthermore, since the data wire 132 is biased high due to the weak pull-up resistor, the master component need not assert any data on the data wire 132 during the preamble phase, thereby reducing power requirement. Additionally, the use of the weak [[week]] pull-up resistor has the advantage of avoiding bus contention and potential electrical issues associated with both the master and slave simultaneously driving the data bus). Note that the preamble could be a sequence of logical zeros (instead of ones) with rest of the frame having interspersed guaranteed ones (instead of zeros), and with the resistor being a weak pull-down resistor, with the same effect. For example, referring to Figure 1, if the supply voltage 141 were a low supply voltage, the resistor 142 would be the pull-down resistor. In the case of a

pull-down resistor, the preamble and the bit 01 would all be logical zeros rather than logical ones as shown.

Please replace paragraphs [0064] and [0065] with the following amended paragraphs:

[0064] In the case of a read operation, the slave component 121 then has the opportunity to pause the frame in cases in which the slave component 121 is not ready to continue at this stage. The slave component asserts the bus hold bit 32 to a logical zero if it is not ready to continue. When ready to continue, the slave component 121 asserts a logical one if it is ready to proceed thereby giving [[given]] the master component 110 notice that the slave component is ready to continue. This provides the slave component 121 with an option to pause the frame when the slave component is not ready to continue for the time being. An additional pausing option available to the slave component is described below with respect to the acknowledgement bit. In the case of a write operation, the bus hold bit 32 is a guaranteed logical one. The bus hold bit 32 is an example of the bus hold field 208 of Figure 2.

[0065] In the case of a read operation, after the slave component 121 transmits the bus [[bit]] hold bit 32, the slave component 121 transmits the eight most significant bits followed by a guaranteed zero bit. In the case of a write operation, after the master component 110 transmits the bus [[bit]] hold bit 32, the master component 110 transmits the eight most significant bits followed by the guaranteed zero bit. In either case, the eight most significant bits are represented by bits 31:24, and the following guaranteed zero bit is represented by bit 23.